

CHAPTER

4

Planning Site Navigation

When you complete this chapter, you will be able to:

- Understand navigation principles
- Build navigation schemes that meet the user's needs
- Provide location information
- Use hypertext linking creatively
- Use graphics for navigation and linking

The free-flowing nature of information in a nonlinear hypertext environment can be confusing to navigate. Help your users find content easily rather than making them hunt through a maze of choices. Reassure your users by letting them know where they are at all times and where they can go. In this chapter you will learn to build user-focused navigation within the hypertext environment to accomplish these goals.



CREATING USABLE NAVIGATION

The PC Webopaedia defines hypertext as a system “in which objects (text, pictures, music, programs, and so on) can be creatively linked to each other...You can move from one object to another even though they might have very different forms.” The idea of hypertext was envisioned in the 1960s by Ted Nelson, who described it as non-sequential writing in his book, *Literary Machines*. Nelson’s basic idea of connecting content through the use of hypertext linking influenced the creators of the Web. With hypertext-linked content, users can traverse information in any order or method they choose, creating their own unique view.

Hypertext is a rich environment in which to write and structure information. In traditional paper-based media, users navigate by turning pages or referring to a table of contents or index separate from the information they are reading. In a hypertext document, users can connect instantly to related information. Hypertext forms of traditional navigation devices, such as tables of contents and cross-references, can be consistently displayed alongside related content. The user can explore at will, jumping from one point of interest to another. Of course, this capability is dependent on the hypertext author adding enough links to facilitate navigation.

In HTML, hyperlinks are easy to create and add no download overhead when they are text-based, yet many sites fail to provide even the most basic navigation options to the user. When you are planning your site navigation, do not skimp on navigation cues, options, and contextual links. You can add graphics easily to create attractive navigation elements, as most Web sites do. Remember that every graphic you add to your Web site increases the download time for the user. Keep your navigation graphics simple, and reuse the same graphics throughout your Web site. Once the navigation graphics are loaded in the user’s cache, the server does not have to download them again. Use an alternate set of text links in the event that the user cannot or will not view your graphics. You will learn more about this later in this chapter.

Effective navigation includes providing cues to the user’s location, not only links to other pages in the Web site. Provide enough location information to let users answer the following navigation questions:

- Where am I?
- Where can I go?
- How do I get there?
- How do I get back to where I started?

To allow users to answer these questions, provide the following information:

- Let the users know their current page and what type of content they are viewing.
- Let users know where they are in relation to the rest of the Web site.
- Provide consistent, easy-to-understand links.
- Provide alternatives to the browser’s Back button that lets users return to their starting point.

LOCATING THE USER

Figure 4-1 shows a page from the Snap Web site that displays a number of user-orienting features.

FIGURE 4-1
Providing user
location cues



The navigation cues on this page give users many options without disorienting them. A linked path at the top of the page shows the user's location within the Web hierarchy. Users can see they are on the Web Authoring page, which is contained in the Web Building section, which is part of the Internet section, and so on. Users can click any of the links in the path to move through the content structure. This location device is especially effective in guiding users who may have arrived at this page from somewhere other than within this Web site. A significant, eye-catching banner also identifies the page, and meaningful headings break up the content. Using these navigation devices, users can choose to jump directly to a page, search for information, or move back up through the information hierarchy.

Figure 4-2 shows a section page from the AOL Web site, which also offers the same types of helpful navigation devices as the page from the Snap Web site.

FIGURE 4-2
Variety of
navigation cues



The location information on this page includes a “You are here” navigation path that tells the user the page they are viewing and the path back up through the hierarchy. The navigation path indicates the current page is the Pictures & Albums page, which is a subtopic of the Web Centers page. Web Centers is one of the main topics on the Home page. A banner in large, dark text identifies the page. Even though this page contains many choices, logical headings break up the lists of single word, meaningful links.

LIMITING INFORMATION OVERLOAD

Many Web sites tend to present too much information at one time. Lengthy files that scroll on and on, or arrays of links and buttons can frustrate and overwhelm the user. You can limit information overload in the following ways:

- Create manageable information segments — Break your content into smaller files, and then link them together. Provide logical groupings of choices. Notice how both pages in Figures 4-1 and 4-2 break up content with logical headings.
- Control page length — Do not make users scroll through never-ending pages. Long files also can mean long downloads. Provide plenty of internal links to help users get around or keep the pages short. You can judge your page length by pressing the Page Down key. If you have to press Page Down more than two or three times to move from the top to the bottom of your page, break up the file.
- Use hypertext to connect facts, relationships, and concepts — Provide contextual linking to related concepts, facts, or definitions, letting the users make the choices they want. Know your material, and try to anticipate the user’s information needs.

USING TEXT-BASED NAVIGATION

Text-based linking often is the most effective way to provide navigation on your site. It can work in both text-only and graphical browsers, and does not depend on users choosing to turn on graphics in their browser. Although you may want to use linked graphics for navigation, include a text-based set of links as an alternate means of navigation. The HoefferType Foundry Web site in Figure 4-3 uses identical text-based and graphic links.

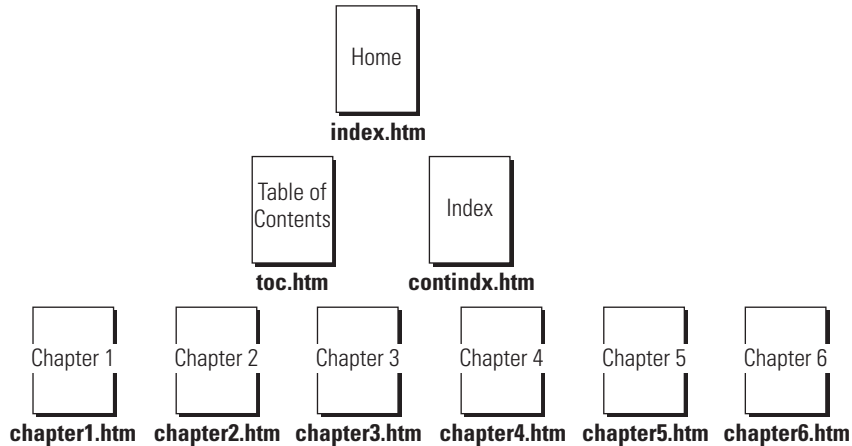
FIGURE 4-3
*Identical text-based and
graphic links*



The graphic links on this page are listed in a column along the left margin so they stand out. They also are separated visually from the text-based links, which are easy to find under the main heading. Instead of seeming repetitive, the text links provide an alternate means of navigation and a quick list of contents for users who cannot view graphics or who are not willing to wait for the images to download.

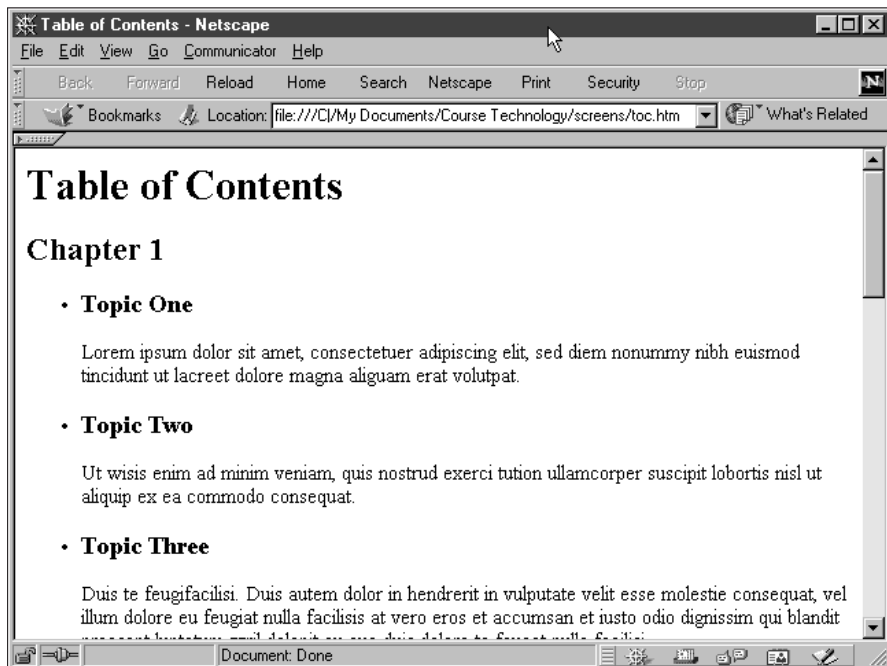
Figure 4-4 provides an example of effective hypertext linking using a collection of HTML documents that includes a Home page, Table of Contents page, Index page, and Chapter pages.

FIGURE 4-4
Sample Web site



The focus is on the Table of Contents page, *toc.htm*, and how it relates to the rest of the content in the collection. In the hypertext environment, the user should be able to select links in the table of contents to jump to any document in the collection. Figure 4-5 shows *toc.htm* in the browser.

FIGURE 4-5
toc.htm in the browser

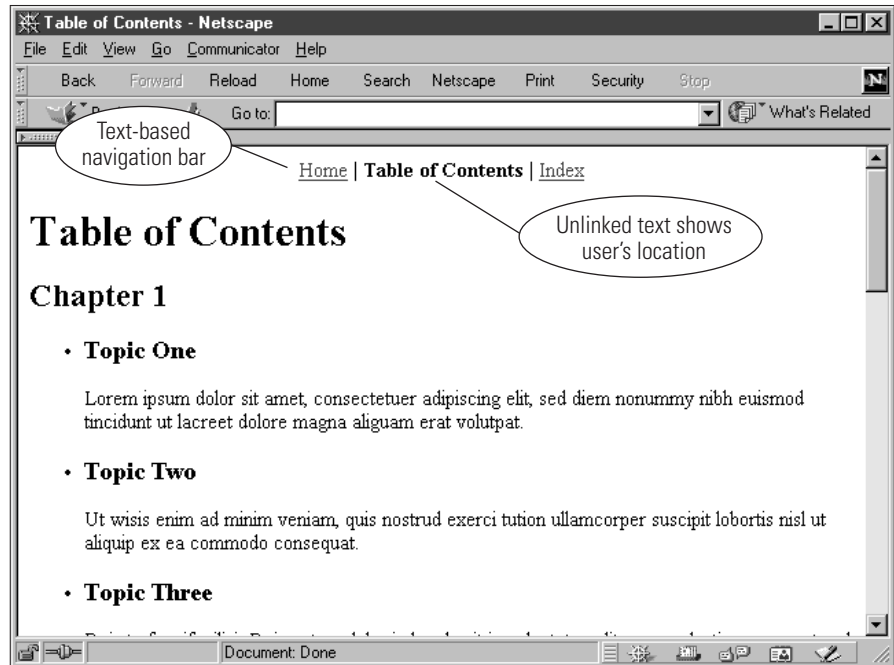


Currently this sample page contains no hypertext links. Users expect a table of contents to list the main topics covered in the document or Web site and to indicate how to find those topics. The following sections describe a variety of linking options that will accommodate different paths through the information.

LINKING WITH A TEXT-BASED NAVIGATION BAR

The Table of Contents page must link to the other main pages of the Web site. Users then can click the links to go directly to the pages they want. You can meet this need by adding a simple text-based navigation bar. Figure 4-6 shows the Table of Contents page with a navigation bar at the top.

FIGURE 4-6
*Adding a text-based
navigation bar*



You can add this navigation bar to the top of every page in the collection, and you may want to add one at the bottom of every page as well. In the Web page illustrated in Figure 4-6, viewers can click the Home link in the navigation bar to go to the Web site's Home page and the Index link to see an index of keywords they can use to find information.

Because this is the Table of Contents page, the Table of Contents text is not a hypertext link but is bold to designate the user's location. The code looks like this:

```
<DIV ALIGN=CENTER>
<A HREF="index.htm">Home</A> | <B>Table of Contents</B> |
<A HREF="contindx.htm">Index</A>
</DIV>
```

The <DIV> element centers the navigation bar on the page, taking the place of the more common, but deprecated, <CENTER> element. Spaces and the vertical bar character separate the links. Notice that the HTML files referenced in the HREF attributes have no path attributes; they reside in the same directory as toc.htm.

TIP

In this and some other code examples in this book, you will notice the use of the `<DIV>` tag. `<DIV>` indicates a logical division in a document. You can use it to specify characteristics for the division, such as `ALIGN=CENTER`, and especially with Cascading Style Sheets. Use `<DIV>` instead of `<P>` in the navigation bar because `<DIV>` has no leading or trailing vertical white space like `<P>`. Using `<DIV>` lets the browser display the navigation bar as close to the top of the page as possible. In general, you can replace the deprecated `<CENTER>` tag with `<DIV ALIGN=CENTER>` in almost every situation.

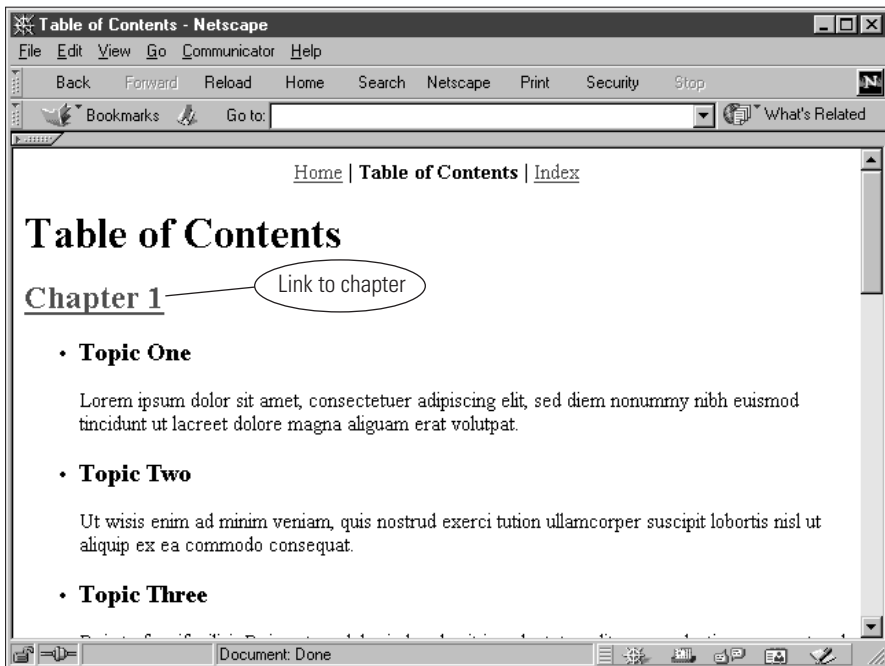
LINKING TO INDIVIDUAL FILES

While the navigation bar lets users access the main pages in the Web site, the table of contents lets users access the exact content they want. The Table of Contents page therefore needs links to the individual chapter files in the Web site. For example, you can add separate links to each chapter heading in the file, as shown in Figure 4-7.

This linking method lets the users scroll through the table of contents to scan the chapters and topics, and then jump to the chapter they want. The link colors—by default, blue for new and purple for visited—allow the user to keep track of which chapters they already have visited. The chapter links are standard hypertext:

```
<H2><A HREF="chapter1.htm">Chapter 1</A></H2>
```

As this example shows, remember always to make `<A>` the innermost set of tags to avoid extra space in the hypertext link.

**FIGURE 4-7**

Adding individual file links

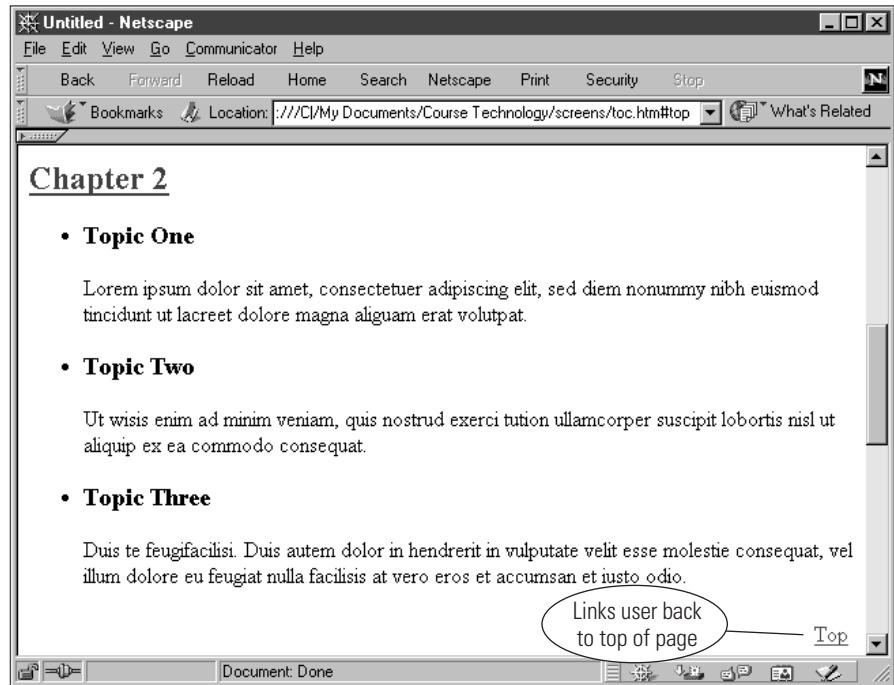
LINKING TO DOCUMENT FRAGMENTS

Besides linking to documents, you also can add links for navigating within the table of contents itself. In the sample Web page illustrated in Figure 4-7, users can drag the scroll bar to view the rest of the table of contents, and then click the chapter link to view a chapter. If they scroll to the bottom of the table of contents and do not click a chapter link, they must scroll back up to the top to use the navigation bar or to choose a different chapter. Instead of making users

scroll, let them navigate the document using hypertext. You can add a top link that lets users return to the top of the page from many points within the file.

Figure 4-8 shows a Top link in the middle of the Table of Contents page.

FIGURE 4-8
Adding a Top link



TIP

The **fragment identifier** must contain at least one character to work properly in all browsers. Constructing an empty fragment identifier such as the one below is not a valid link destination.

If you have no characters to bracket with `<A>` tags, use a non-breaking space character entity, as in the example below:

```
<A NAME="top">
&nbsp;</A>
```

This will work as an invisible link destination.

The Top link lets the user jump back to the top of the file. This requires two `<A>` anchor elements: one uses the NAME attribute to name a **fragment** of the document; the other targets the fragment name in the HREF attribute. The two `<A>` elements that create a Top link look like this:

```
<A NAME="top">any character</A>
```

This is the destination anchor of the Top link. The value of the NAME attribute can be any alphanumeric combination you choose, so make it something meaningful to indicate the top of the page.

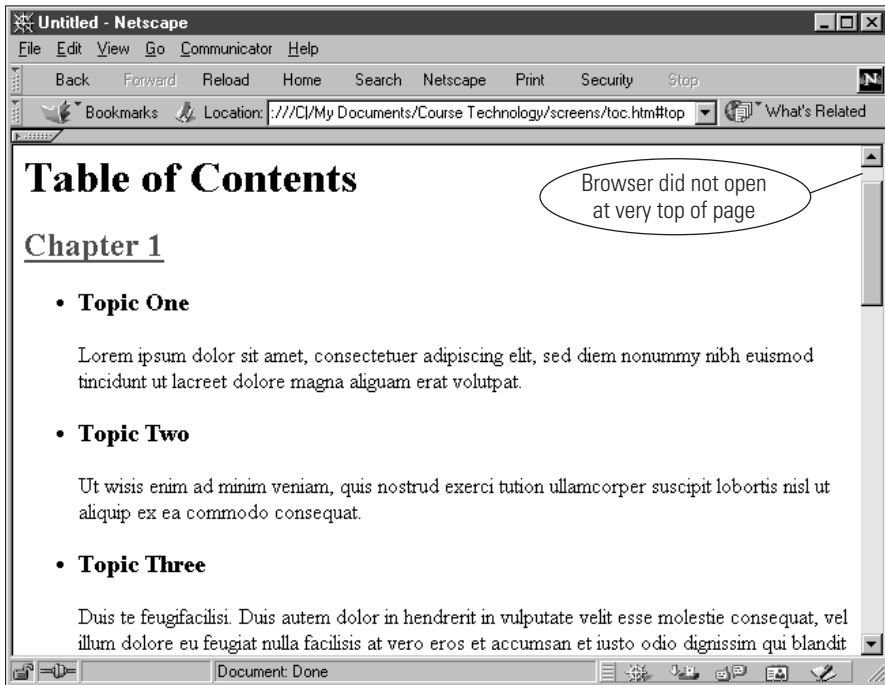
Add a standard anchor tag later in the file where you want the link to appear. This is the source anchor. Reference the destination fragment by using the number sign (#) in the HREF attribute like this:

```
<A HREF="#top">Top</A>
```

Where you place the fragment identifier in your code is important. When the user clicks the word Top, the browser opens to the exact spot in the file designated

by the NAME attribute. Figure 4-9 shows the result of users clicking the Top link when the fragment identifier is placed incorrectly.

FIGURE 4-9
*Incorrect placement of
fragment identifier*



Compare this screen to the one in Figure 4-7. Due to the placement of the fragment identifier, the navigation bar does not appear because the browser is not scrolled entirely to the top of the page. Look at the code from the top of the toc.htm file:

```
<BODY>
<DIV ALIGN=CENTER><A HREF="index.htm">Home</A> | <B>Table
of Contents</B> | <A HREF=" contindx.htm ">Index of
Contents</A></DIV>
<BR>
<A NAME="top"><H1>Table of Contents</H1></A>
```

The placement of the fragment identifier is near the <H1>, but it needs to be moved to the top of the file. The navigation bar is the first item in the file and already contains anchor elements, so it cannot be nested within another anchor element. The following code will not work because <A> tags cannot be nested.

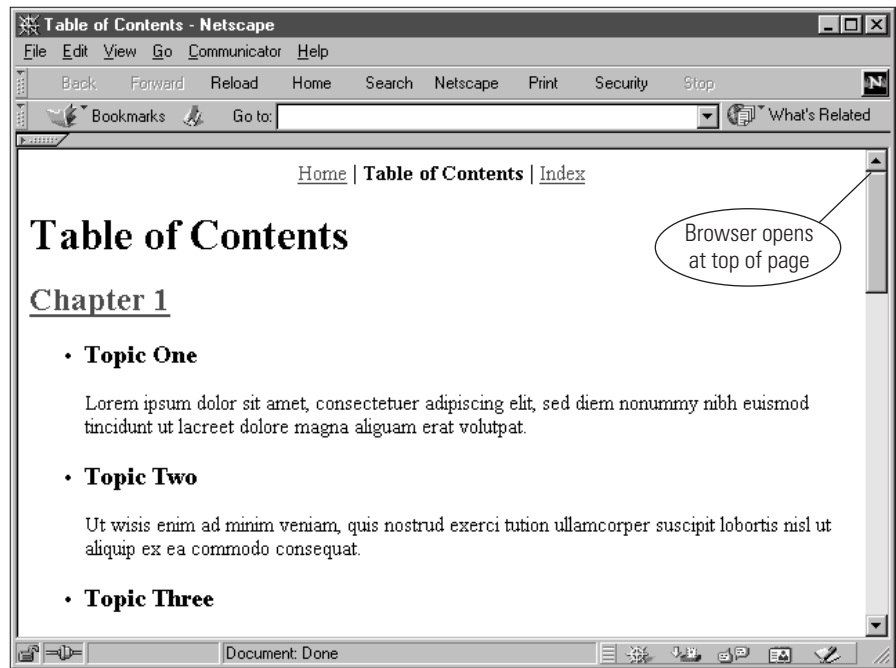
```
<BODY>
<A NAME="top"><DIV ALIGN=CENTER><A HREF="index.htm">Home</
A> | <B>Table of Contents</B> | <A HREF=" contindx.htm ">
Index</A></DIV></A>
<BR>
<H1>Table of Contents</H1>
```

The solution to this problem is to use the NAME attribute in the anchor tag. This code makes the link text, Home, both a source and destination anchor:

```
<BODY>
<DIV ALIGN=CENTER><A HREF="index.htm" NAME="top">Home</A>
| <B>Table of Contents</B> | <A HREF=" contindx.htm ">
Index of Contents</A></DIV>
<BR>
<H1>Table of Contents</H1>
```

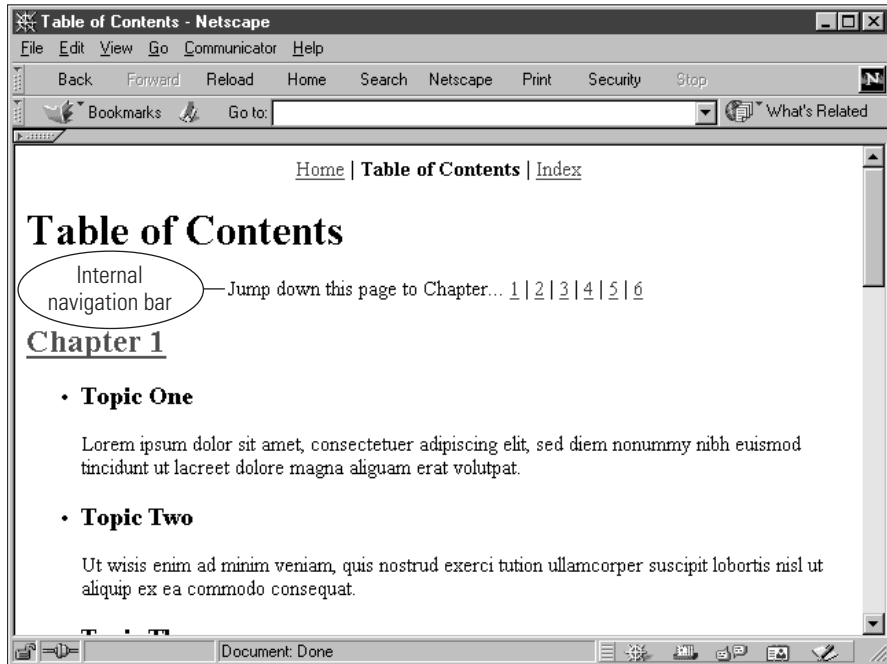
With this code in place, the browser will open to the very top of the page as shown in Figure 4-10.

FIGURE 4-10
*Correct placement of
fragment identifier*



You can use additional fragment identifiers to add more user-focused navigation choices. Besides including links users can click to return to the top of the Table of Contents page, you can add an internal navigation bar that lets users link to topics within the list of chapters. Figure 4-11 shows the addition of an internal navigation bar.

FIGURE 4-11
Adding an internal
navigation bar to the
table of contents



When users click one of the linked chapter numbers, they jump to the specific chapter they want to view within the table of contents.

Here is the code for the navigation bar:

```
<DIV ALIGN=CENTER>
Jump down this page to Chapter... </FONT><A HREF=
#chapter1">1</A> | <A HREF="#chapter2">2</A> | <A HREF=
#chapter3">3</A> | <A HREF="#chapter4">4</A> | <A HREF=
#chapter5">5</A> | <A HREF="#chapter6">6</A> </DIV>
```

The <DIV> element centers the navigation bar on the page. The HREF attributes point to fragment identifiers. To ensure that this internal linking works properly, add the fragment identifiers to the file for each chapter heading. Remember that these headings are linked already to each of their respective chapter files. The existing link for the Chapter page looks like this:

```
<H2><A HREF="chapter2.htm">Chapter 2</A></H2>
```

Because the text already is contained in an anchor element, use the NAME attribute to name the fragment:

```
<A HREF="chapter2.htm" NAME="chapter2">Chapter 2</A>
```

Now users can click one of the links in the internal navigation bar and jump directly to that place in the file.

LINKING TO EXTERNAL DOCUMENT FRAGMENTS

Now that the sample Web page provides the user with plenty of internal navigation choices, re-examine how the table of contents is linked to the chapter files. Currently, each chapter has one link in the table of contents. Users click the chapter link and the browser opens the chapter file at the top. Each chapter also contains multiple topics. You can let users jump to the exact topic they want within each chapter. This will require marking up the chapter documents with topic fragment identifiers. The following code shows the topic1 fragment in chap2.htm:

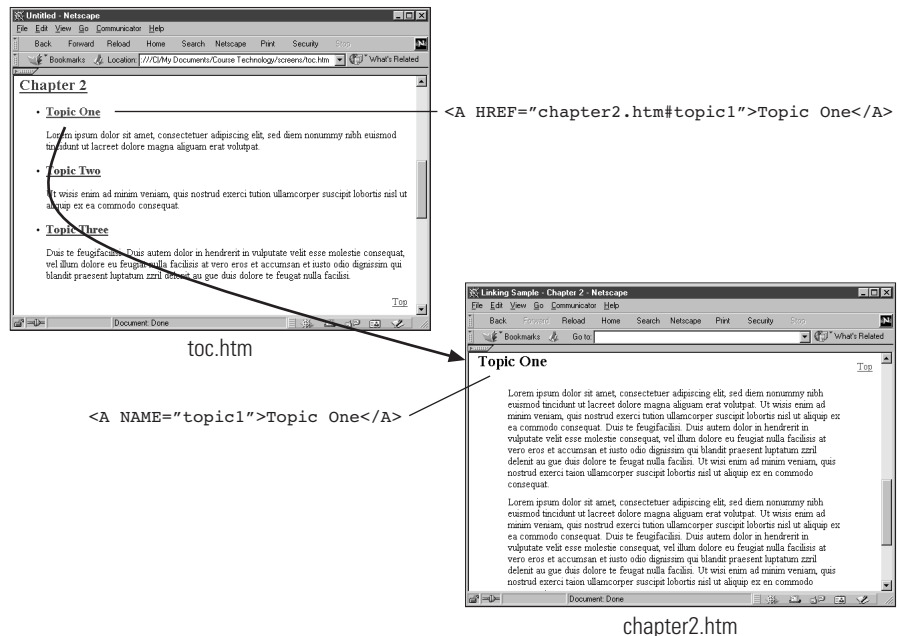
```
<A NAME="topic1">Topic One</A>
```

To reference this in the table of contents toc.htm, use the following code:

```
<A HREF="chapter2.htm#topic1">Topic One</A>
```

When users click the topic linking the table of contents, the browser opens the destination file and displays the fragment (Figure 4-12).

FIGURE 4-12
Linking to an external fragment



TIP

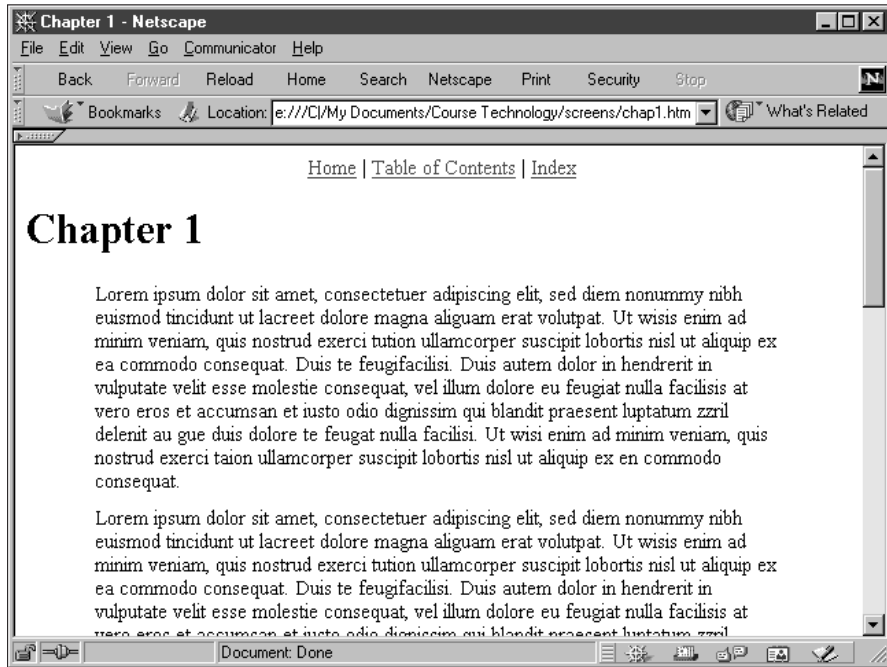
HTML 4.0 adds the ID attribute. You can use ID to identify a fragment, just as NAME does. The great feature of ID is that you can use it with almost any element. If you want to identify an <H1> heading as a fragment, add the ID attribute to the <H1> tag:

```
<H1 ID="topic1">Topic One</H1>
```

The downside of ID is its lack of support in many browsers, including Netscape 4.08. Internet Explorer 5.0 does support the ID attribute. Be certain to test in multiple browsers if you choose to use ID.

Now that the table of contents is linked fully to the rest of the site, examine the individual chapter files to see if they can use any additional links. Figure 4-13 shows the top of Chapter 1 in the browser.

FIGURE 4-13
Chapter1.htm in the
browser



Adding Page Turners

Each chapter file currently contains a navigation bar and fragment identifiers for each topic within the chapter. In this page collection, the user can jump to any file and topic within a file, though some users may want to read the pages sequentially. You can enhance the navigation bar in the chapter pages by adding page-turner links. Page turners let you move either to the previous or next page in the collection. These work well in a linear structure of pages as shown in Figure 4-14.

Note that Chapter 1 includes the table of contents as the previous page, while Chapter 6 uses the index as the next page. The code for the new navigation bar looks like this:

```
<DIV ALIGN=CENTER>
<A HREF="index.htm" NAME="top">Home</A> | <A HREF="toc.htm"
">Table of Contents</A> | <A HREF="toc.htm">Previous</A> |
<A HREF="chapter2.htm">Next</A> | <A HREF="index.htm">
Index</A>
</DIV>
```

Figure 4-15 shows the enhanced navigation bar at the top of the page.

FIGURE 4-14
Sequential page turning

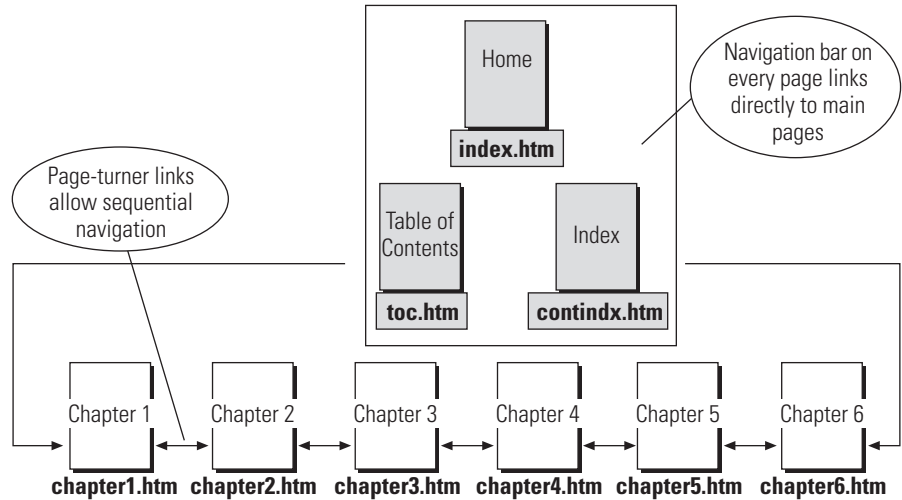
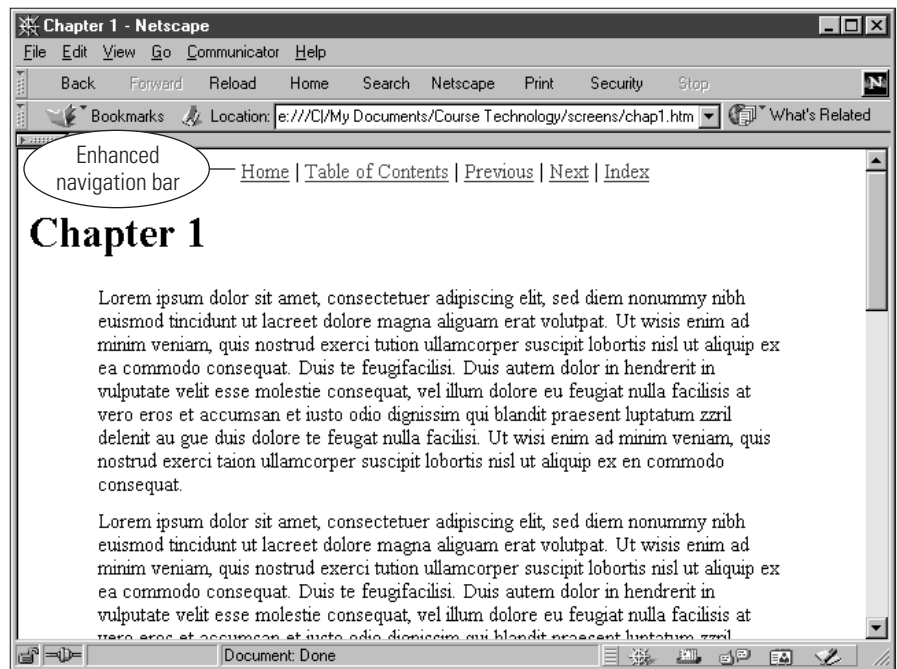
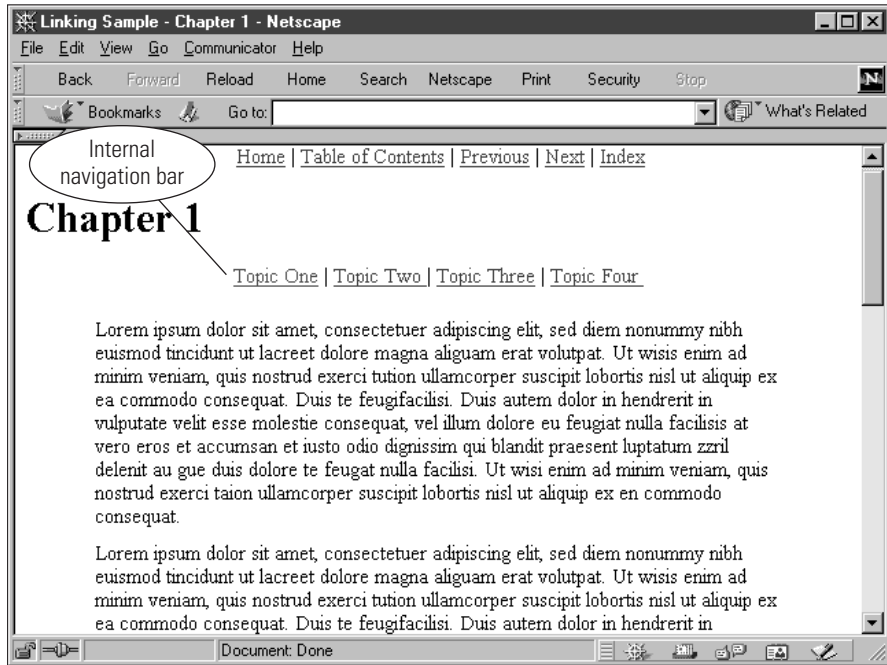


FIGURE 4-15
Adding page turners to the navigation bar



Finally, you can make double use of the fragment identifiers that name each topic within the chapter by adding an internal navigation bar to each chapter. This is the same type of navigation you added to the table of contents, but it helps users navigate from topic to topic within a chapter. Figure 4-16 shows the new navigation bar.

FIGURE 4-16
Adding an internal
navigation bar



The code for the chapter navigation bar looks like this:

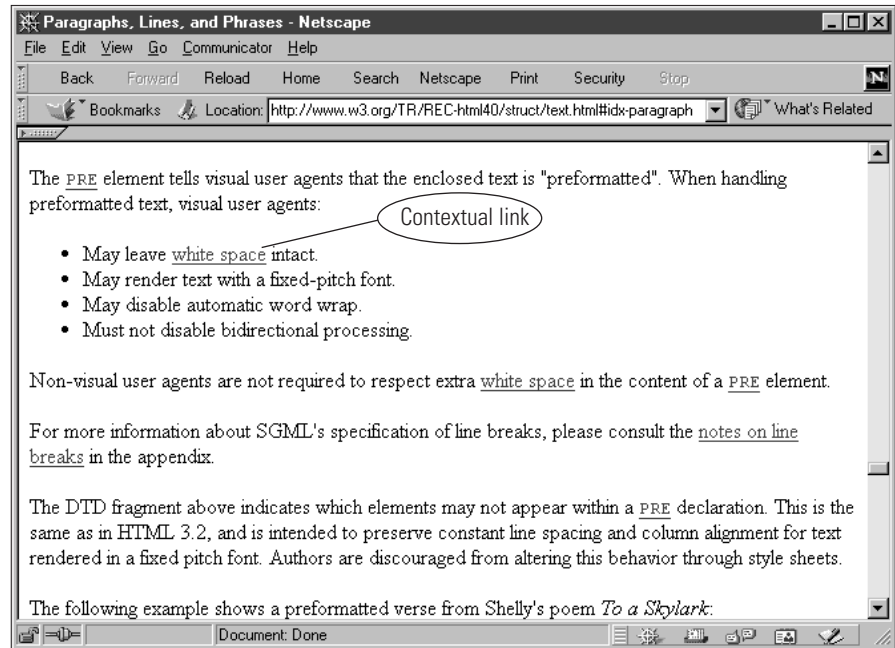
```
<P ALIGN=CENTER>
  <A HREF="#topic1">Topic One</A> | <A HREF=" topic2.htm">
Topic Two</A> | <A HREF="topic3">Topic Three</A> |
<A HREF="topic4">Topic Four</A> </P>
```

Using the `<P>` element instead of `<DIV>` provides extra white space above and below the navigation bar. You can place one of these navigation bars in each chapter and adjust the number of topic links appropriately.

ADDING CONTEXTUAL LINKING

Many Web sites fail to use one of the most powerful hypertext capabilities—the contextual link. **Contextual links** allow users to jump to related ideas or cross-references by clicking the word or item that interests them. These are links that you can embed directly in the flow of your content by choosing the key terms and concepts that you anticipate your users will want to follow. Figure 4-17 shows a page from the World Wide Web Consortium HTML specification that contains contextual linking.

FIGURE 4-17
Contextual linking



Note the links within the lines of text, which let the user view related information in context. For example, as users read the first bulleted item, “May leave white space intact,” they can click the white space link to see a definition of that term. Including the link within a line of text is more effective than including a list of keywords because users can see related information within the context of the sentence they are reading. Users also can see that repeated words are linked no matter how many times they appear within the browser window, offering users the opportunity to access additional information at any time.

You can choose from a variety of navigation options to link a collection of pages together. The sample Web pages in this section demonstrated the following text-based linking actions:

- To main pages (home, table of contents, index)
- To the top of each chapter
- Within the Table of Contents page to chapter descriptions
- From the Table of Contents page to specific topics within each chapter
- Between previous and next chapters
- Within chapter pages of each topic
- To related information by using contextual links

Use as many of these options as necessary, but remember to view your content from the user’s perspective. Anticipate and use enough navigation options to allow easy, but not confusing, access to your content.

USING GRAPHICS-BASED NAVIGATION

Similar to most Web site designers, you probably want to use graphics for some of your navigation cues. The ability to use graphics is one of the most appealing aspects of the Web, but too many graphics used inconsistently will confuse the users. To make sure your navigation graphics help rather than hinder your users, use them consistently throughout your Web site. Standardize your navigation graphics for the following reasons:

- To provide predictable navigation cues for the user — Once users learn where to find navigation icons and how to use them, they anticipate them on every page. Consistent placement and design also build trust in your users and help them feel confident that they can find the information they want.
- To minimize download time — Once the graphic is downloaded, the browser will retrieve it from the cache for subsequent pages rather than downloading it every time it appears.

USING TEXT IMAGES FOR NAVIGATION

Navigation graphics on the Web come in every imaginable style. Many sites use text images, rather than HTML text, for navigation graphics. Text images are text created as graphics, usually as labels within the graphic. Many Web designers prefer text images because they have more typeface and design choices when creating their own graphics.

Figure 4-18 shows the top navigation bars from a standard page of *The Sydney Morning Herald* Web site. Notice that this text uses graphics instead of HTML text.

The navigation bar builds the page name, Web site name, main sections, and page turners into a unified graphic that serves as the banner for the top of the page. The banner creates a unified look to all of their content, while providing a variety of useful navigation choices. Note that what appears to be a single graphic actually is composed of different graphics held together by a table, a very common technique which you will learn more about in Chapter 7.

USING ICONS FOR NAVIGATION

Figure 4-19 shows the main page navigation bar from Travelocity, a travel Web site.

FIGURE 4-18
The Sydney Morning Herald's navigation bar

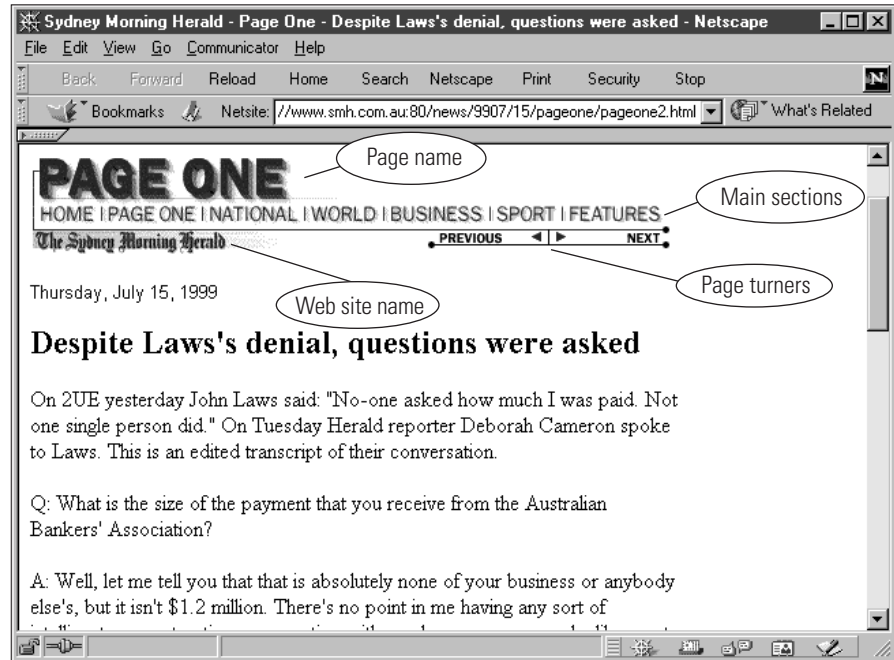
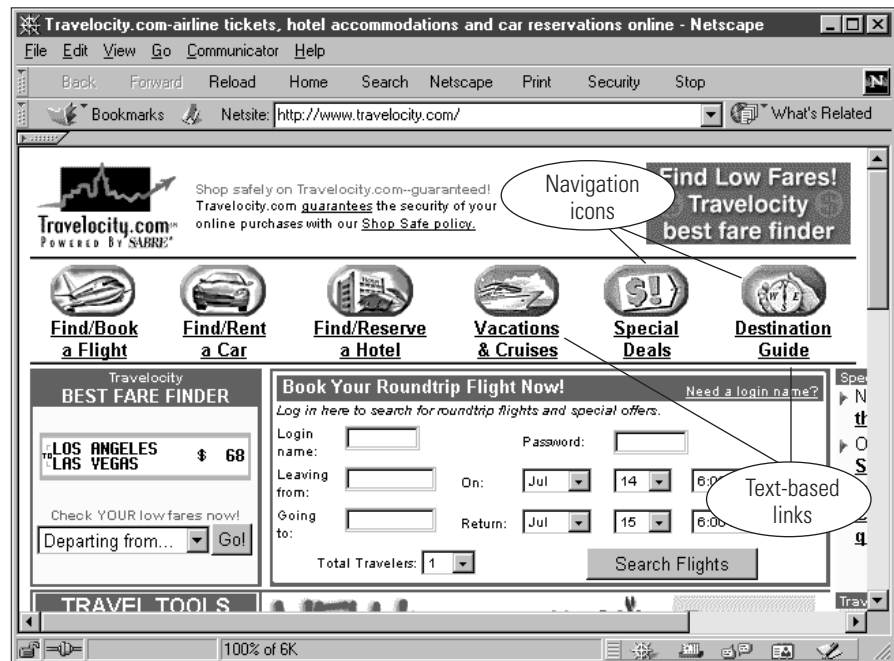
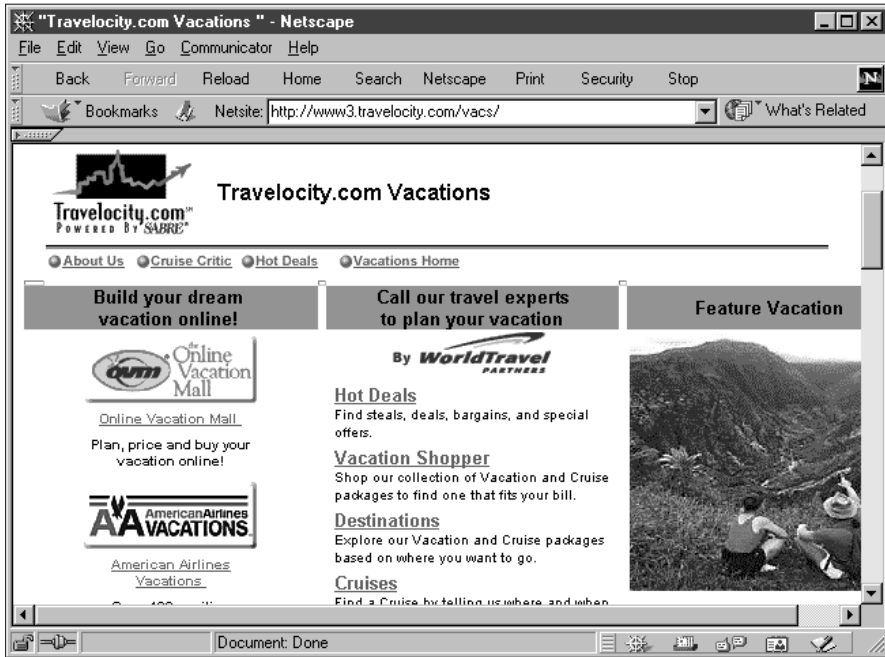


FIGURE 4-19
Icons for navigation



The navigation bar for this Web site uses icons and text-based links. The text-based links serve a dual purpose: if the graphics for the icon do not appear, the user still can navigate, and the text describes each icon. It would be ideal if the design carried these icons through to each of the destination pages, but unfortunately they only appear on the main page of the Web site, thus missing an opportunity to use consistent identifying navigation graphics. Note that the Vacations & Cruises page (Figure 4-20) shows no trace of the navigation bar.

FIGURE 4-20
Missing navigation bar



The text labeling each image in the Travelocity Web site points out one of the main problems with icons—not everyone agrees on their meaning. Especially with a worldwide audience, you never can be sure exactly how your audience will interpret your iconic graphics. This is why so many Web sites choose text-based links, even if they are text as graphics. If you do use navigation icons, be sure to define them; that is, possibly use a table that lists each icon and describes its meaning. Figure 4-21 shows a page from a student project Web site that clearly explains the Web site's navigation icons.

No matter what type of graphics you choose as icons, make sure that your users will understand their meaning. Test your navigation graphics with people from your target audience and ask them to interpret the icons and directional graphics you want to use. The most obvious type of graphics to avoid are symbols that are culturally specific, especially hand gestures (such as thumbs-up) that may be misinterpreted in other cultures. Other graphics, such as directional arrows, are more likely to be interpreted correctly. Figure 4-22 shows an icon-based navigation bar that includes universal previous, top, and next links.

FIGURE 4-21
Clear definition of
navigation icons

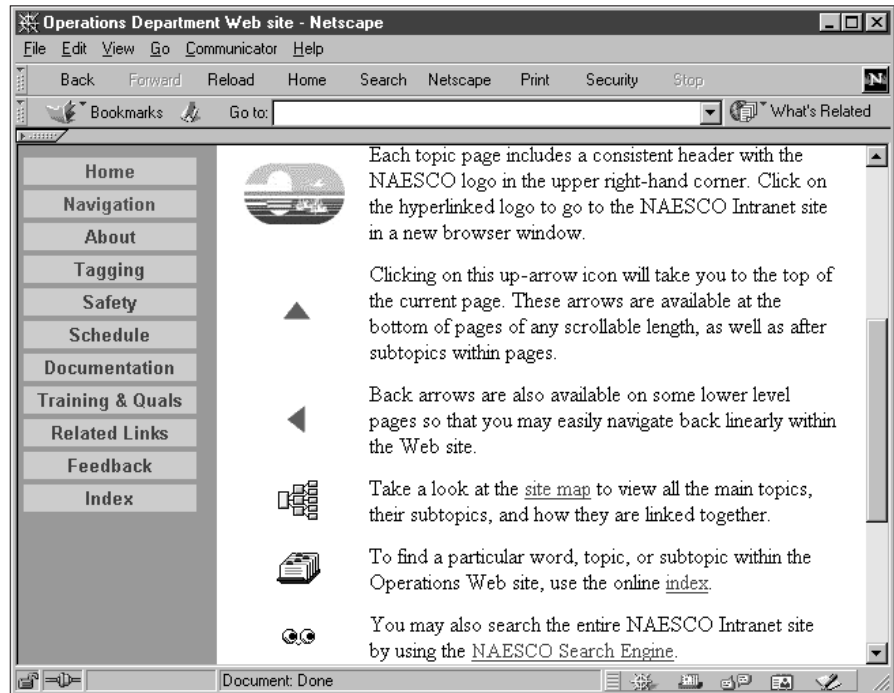
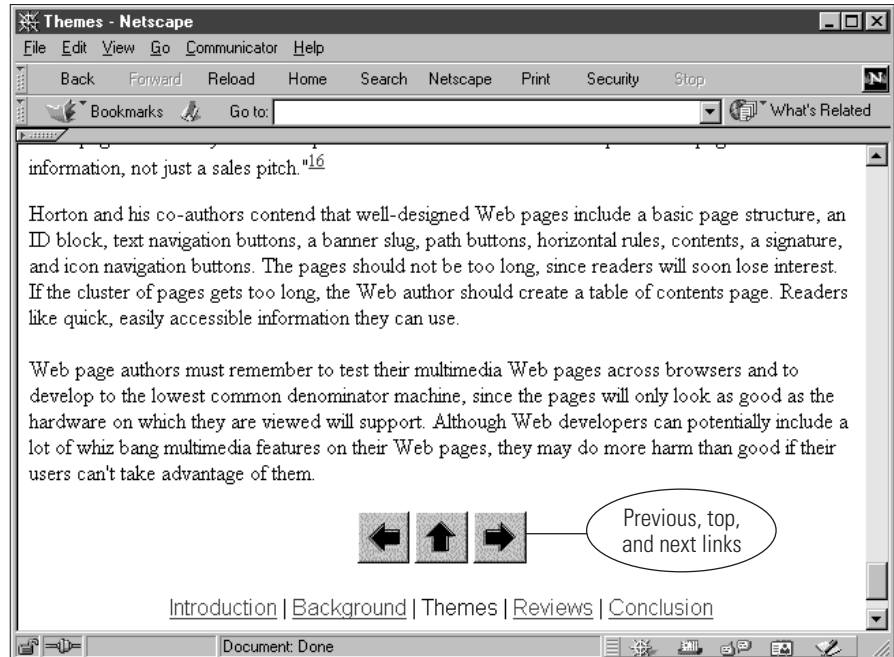
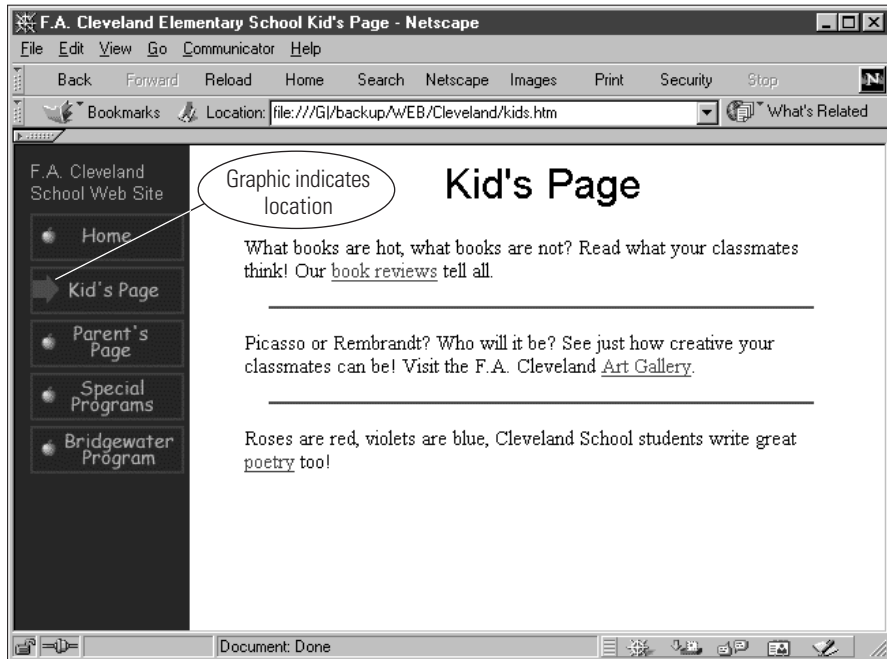


FIGURE 4-22
Universal navigation
graphics



You also can use navigation graphics to indicate location within a site. For example, you can change the color or shading to indicate which page the user currently is viewing. At the F. A. Cleveland Elementary School Web site, illustrated in Figure 4-23, the navigation graphic includes an arrow to indicate which section of the site the user currently is viewing.

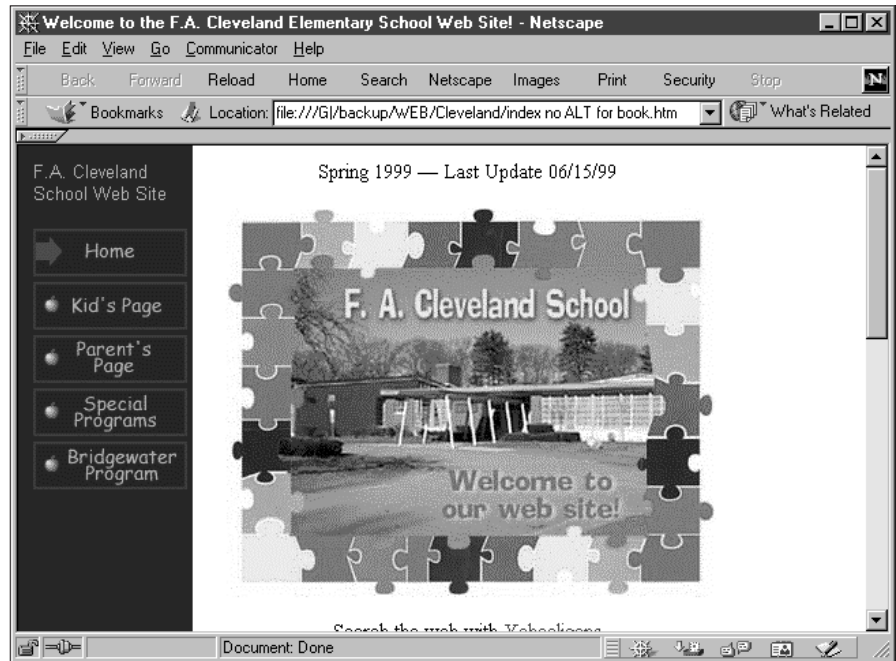
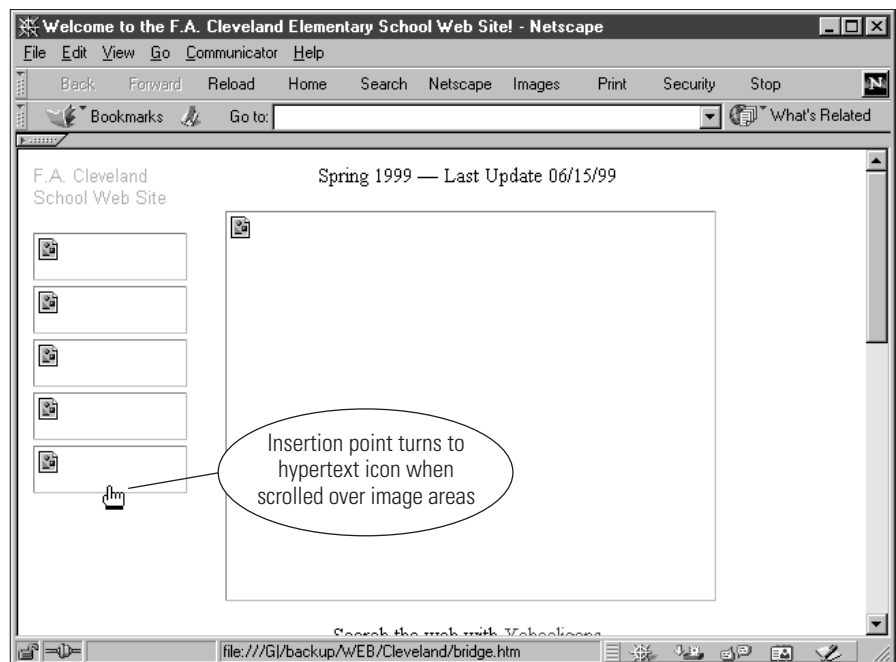
FIGURE 4-23
Navigation graphic indicates location



USING THE ALT ATTRIBUTE

As you read earlier, you should provide alternate text-based links in addition to graphical links. You can do so by including an ALT attribute in the IMG tag of the HTML code for the graphic. Repeating navigation options ensures you meet the needs of a wide range of users. Some sites choose not to offer a text-based alternative, and this will make it difficult for users who cannot view graphics in their browsers. Figure 4-24 shows the main page of the F. A. Cleveland Elementary School Web site, which consists almost entirely of graphics.

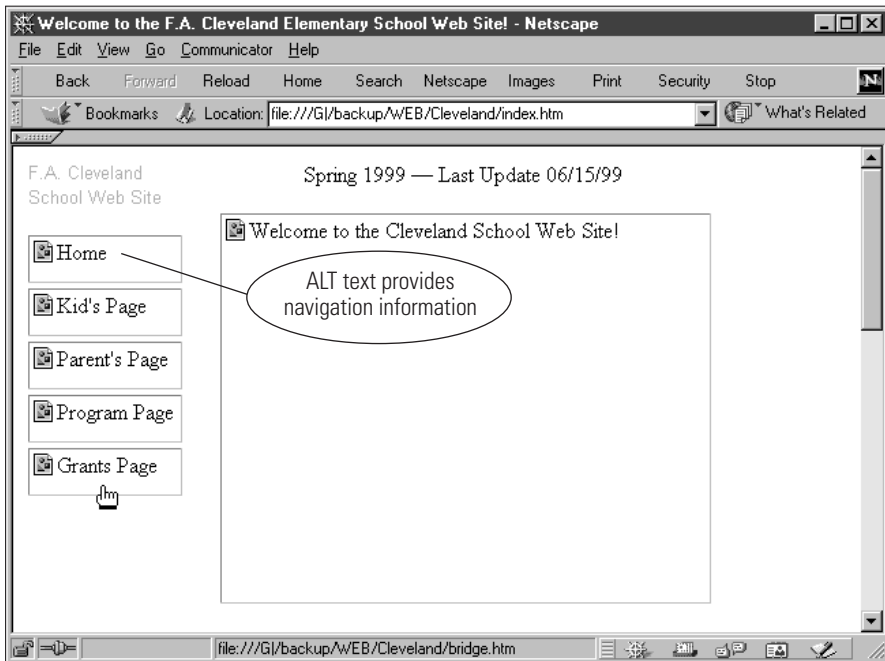
The navigation images in the left column are the only way to navigate the site. Users cannot leave the main page if graphics are turned off or do not download. Figure 4-25 shows the same Web page with images turned off.

FIGURE 4-24*Graphics-only navigation***FIGURE 4-25***No ALT values in tags*

Without the graphics, this site offers no navigation information. By omitting ALT attributes and relying on graphics for navigation, the user cannot effectively navigate the site. Even though graphics are turned off, they retain their hypertext characteristics as shown by the hypertext pointer, but the user does not have any destination information.

By adding descriptive ALT text, non-graphical browser users can navigate your site. Figure 4-26 shows the F. A. Cleveland Elementary School Web site with the ALT attributes added.

FIGURE 4-26
ALT values in
 tags



With the graphics turned off, users still can navigate the Web site because the ALT attribute values appear in the image space. The user finds navigation cues by reading the ALT text and pointing to the image areas to find the clickable spots.

The code for one of these navigation buttons looks like this:

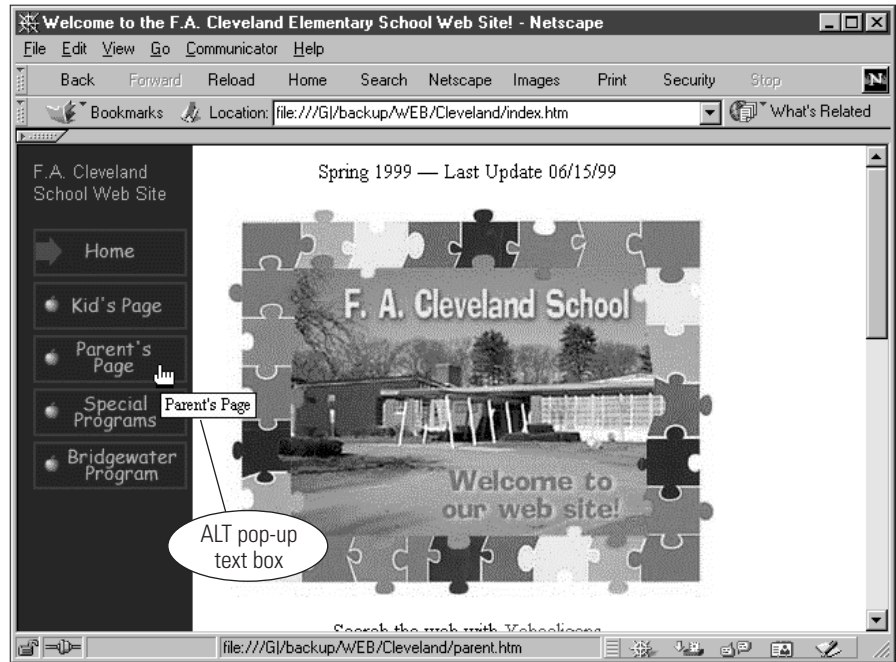
```
<A HREF="parent.htm"><IMG border=0 height=35 src="smparent.gif" width=113 ALT="Parent's Page"></A>
```

Note that you must specify the image width and height in the tag to reserve the image space in the browser.

While the ALT attribute provides valuable navigation information to users who are not displaying graphics or are waiting for graphics to download, it does have some limitations. If your graphics are small, the ALT text may not appear because it needs more room for the text in the graphic area. To solve this problem and to add additional navigation information, both Internet Explorer 4.0/5.0

and Netscape 4.0 display pop-up text boxes if the ALT attribute is included. Figure 4-27 shows an example of a pop-up text box.

FIGURE 4-27
*Value of ALT attribute
displays in pop-up
text box*



When the user points to a graphic, the pop-up text box containing the ALT attribute appears, even if the browser does not display graphics. Not only is this effective if graphics are turned off, but you can add descriptive text that helps the user navigate. For example, pointing to the navigation arrow in Figure 4-27 could display additional navigation information, such as “You are here.”

SUMMARY & REVIEW

Creating usable navigation is the result of working with the power of hypertext and designing for your users' needs. Keep these following points in mind:

- Work from the users' point of view. Think about where users want to go within your Web site, and make it easy for them to get there.
- Add plenty of links to make all areas of your Web site quickly accessible to your users. Link to fragments as well as whole pages. Make it easy to get back to your navigation options.
- In addition to providing links, make sure you provide plenty of location cues to let users know where they are.
- Use text-based navigation bars to link users to other pages in your site. Use other text-based links to help users move through a long page of information or through a table of contents.

- Consider text as an alternative to graphical links. Every additional graphic adds to download time. For the graphics and icons used as navigational links, make sure users will interpret these links correctly by including text as part of the images. Also, be sure to use navigation icons consistently throughout your Web site to provide predictable cues for users and to minimize download time.
- Provide ALT values to your tags to provide alternate navigation options for users.

REVIEW QUESTIONS

1. List three advantages of linking by using text instead of graphics.
2. What four navigation questions does the user have to ask?
3. List three types of navigation cues.
4. List three ways to control information overload.
5. Explain why you would include both graphic and text-based links on a Web page.
6. List two navigation cues you can add to a text-based navigation bar.
7. Which deprecated element can you replace with <DIV ALIGN=CENTER>?
8. Why is it best to make <A> the innermost element to a piece of text?
9. What <A> tag attribute is associated with fragment identifiers?
10. List two ways to break up lengthy HTML pages.
11. What character entity is useful as an invisible link destination?
12. What attribute do you use to make an <A> tag both a source and destination anchor?
13. What HTML 4.0 attribute allows you to create fragment identifiers?
14. How do you link to a fragment in an external file?
15. Page turners work best in what type of structure?
16. What are the benefits of contextual linking?
17. List two reasons for standardizing graphics.
18. What are the benefits of using navigation graphics?
19. What are the drawbacks of using navigation icons?
20. What are the benefits of using the ALT attribute?

PROJECTS

1. Browse the Web and find a Web site that has a successful navigation design. Write a short summary of why the navigation is effective and how it fits the user's needs.
2. Find an online shopping Web site.
 - a. Examine the navigation options and determine whether you think the navigation adds to or detracts from the online shopping experience.
 - b. Describe how to change the navigation to increase its effectiveness.
3. Find an online information resource that likely would be used for research. Examine the navigation options and describe how the navigation aids or hinders the user's information searching process.

4. Use your favorite Web search engine to search for navigation icons.
 - a. Assemble a set of icons that would be suitable for international audiences.
 - b. Assemble a second set of icons that only would be understood by a local population.
5. Browse the Web to find examples of the Web sites that need better navigation options. Using examples from the Web site, describe how you would improve the navigation choices.
6. Browse the Web to find a Web site that uses more than one navigation method and describe why you think this benefits the Web site.
7. Find a site that illustrates a navigation method different from the ones described in the chapter. Describe the navigation method and state why this benefits the Web site.
8. Take an existing paper-based project and turn it into a hypertext document.
 - a. Use a term paper or report from a previous class that you prepared using a word processor and is available in electronic format. Preferably, the document should contain a table of contents and bibliography.
 - b. Convert the document to HTML if the program allows, or save the document as ASCII text and paste it into Notepad or an HTML editor.
 - c. Mark up the document for Web presentation. Include a linked table of contents, topic links, content links, footnote links, and top links. You may find it best to break the single document into a few HTML files and then link them together.
 - d. Test your document in multiple browsers to ensure its portability.
9. This book's companion Web site contains all the HTML files for the sample Web site illustrated in Figure 4-4. Use these sample HTML files to re-create the examples in this chapter or practice different types of hypertext linking.

CASE STUDY

Examine the flowchart you created for your Web site. Consider the requirements of both internal and external navigation. Create a revised flowchart that shows the variety of navigation options you are planning for the Web site.

Using your HTML editor, mark up examples of navigation bars for your content. Make sure your filenames are intact before you start coding. Save the various navigation bars as separate HTML files for later inclusion in your Web pages.

Plan the types of navigation graphics you want to create. Sketch page banners, navigation buttons, and related graphics. Find sources from where you will acquire your navigation graphics. For example, you can use public domain (non-copyrighted) clipart collections on the Web for basic navigation arrows and other graphics.